

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Fabio Casati et al.	§	Art Unit:	2153
		§		
Serial No.:	10/066,098	§	Conf. No.:	6026
		§		
Filed:	January 31, 2002	§	Examiner:	LaShanya Renee Nash
		§		
For:	Dynamic Conversation Logic	§	Atty. Dkt. No.:	10010118-1
	Selection Method and System	§		(HPC.0311US)

**Mail Stop Appeal Brief-Patents**

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

**APPEAL BRIEF PURSUANT TO 37 C.F.R § 41.37**

Sir:

The final rejection of claims 1-3, 5-7, 11-14, 16-18, 27, 28, 30, and 31 is hereby  
appealed.

**I. REAL PARTY IN INTEREST**

The real party in interest is the Hewlett-Packard Development Company, LP. The Hewlett-Packard Development Company, LP, is a limited partnership established under the laws of the State of Texas and has a principal place of business at 11445 Compaq Center Drive West, Houston, TX 77070, U.S.A. (hereinafter "HPDC"). HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware Corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, LLC.

## **II. RELATED APPEALS AND INTERFERENCES**

None.

## **III. STATUS OF THE CLAIMS**

Claims 1-3, 5-7, 11-14, 16-18, 27, 28, 30, and 31 have been finally rejected and are the subject of this appeal.

Claims 4, 8-10, 15, 19-26, and 29 have been cancelled.

## **IV. STATUS OF AMENDMENTS**

No amendment after the final rejection of October 27, 2010 has been submitted. Therefore, all amendments have been entered.

## **V. SUMMARY OF THE CLAIMED SUBJECT MATTER**

The following provides a concise explanation of the subject matter defined in each of the independent claims involved in the appeal, referring to the specification by page and line number and to the drawings by reference characters, as required by 37 C.F.R. § 41.37(c)(1)(v). Each element of the claims is identified by a corresponding reference to the specification and drawings where applicable. Note that the citation to passages in the specification and drawings for each claim element does not imply that limitations from the specification and drawings should be read into the corresponding claim element. Note also that the cited passages are provided as examples, as other passages in the specification or drawings not cited may also be relevant to the corresponding claim elements.

Independent claim 1 recites a method for selecting a conversation logic (Spec., p. 8:21-26) at run-time for a workflow definition (Fig. 2:204) that includes at least one node with no hard-coded conversation logic (Spec., 18:4-6), the method comprising the steps of:

- a) maintaining (Fig. 4:410) a conversation logic repository (Fig. 2:211) that includes plural conversation logic (Spec., p. 8:21-26), wherein each of the plural conversation logic is external to the workflow definition, and wherein each of the plural conversation logic specifies a corresponding set of operations to be performed on a respective service (Spec., 9:3-10);
- b) when executing the node with no hard-coded conversation logic, dynamically discovering (Fig. 4:420), by a computer, a service associated with the node with no hard-coded conversation logic, wherein the discovered service is selected from among plural services (Spec., 18:4-8; 19:13-20:2);
- c) selecting (Fig. 4:430) one of the plural conversation logic in the conversation logic repository based on the discovered service (Spec., 10:20-21; 11:15-16; 18:9-13; 19:21-20:2); and
- d) dynamically plugging (Fig. 4:440) in the determined selected conversation logic into the node at run time in the computer, wherein the run time is a time during which the node with no hard-coded conversation logic is being executed (Spec., 11:16-18; 18:4-20:2).

Independent claim 3 recites a method for selecting a conversation logic (Spec., 8:21-26) at run-time comprising the steps of:

maintaining (Fig. 4:410) a conversation logic repository (Fig. 2:211) that includes plural conversation logic, wherein each of the plural conversation logic specifies a corresponding set of operations to be performed on a respective service (Spec., 9:3-10);

at run-time, sending a service selection query to an electronic services platform (Fig. 3:330) or other service broker, wherein the service selection query is for selecting a service from among plural services, wherein the run-time is a time during which a node of a workflow definition is being executed, where the node is with no hard-code of conversation logic (Spec., 11:6-7, 11-13; 18:4-20:2);

receiving, by a computer, a returned service identifier corresponding to the selected service (Spec., 11:15-16; 18:9-11);

selecting (Fig. 4:430), by the computer, a conversation logic from among the plural conversation logic in the conversation logic repository based on the returned service identifier (Spec., 10:19-21; 18:9-13; 19:21-20:2); and

dynamically plugging (Fig. 4:440) in the selected conversation logic into the node with no hard-coded conversation logic at the run-time (Spec., 11:16-18; 18:4-20:2).

Independent claim 11 recites a system for dynamically selecting a conversation logic (Spec., 8:21-26) at run-time for a workflow definition (Fig. 2:204) that includes at least one node with no hard-coded conversation logic (Spec., 18:4-6) comprising:

- a workflow engine (Fig. 3:310; Fig. 10:1000) for processing a workflow definition (Spec., 11:2-4, 9-13; 16:17-25);

- a conversation logic repository (Fig. 2:211) that includes plural conversation logic that are external to the workflow definition, wherein each of the plural conversation logic specifies a corresponding set of operations to be performed on a respective service (Spec., 8:21-26; 9:3-10);

- an engine (Fig. 3:330) configured to select (Fig. 4:430) one of plural services for execution of the node with no hard-coded conversation logic (Spec., 10:20-21; 11:15-16; 18:9-13; 19:21-10:2); and

- a dynamic conversation logic selection mechanism (Fig. 2:210) configured to receive a service identifier that is associated with the selected service at run-time, and based on the service identifier to select a conversation logic from the plural conversation logic for interacting with the selected service at the run-time (Spec., 9:1-7; 10:19-21; 11:15-16; 18:9-13; 19:21-20:2), and

- wherein the dynamic conversation logic selection mechanism (Fig. 2:210) is configured to further dynamically plug in the selected conversation logic into the node at the run-time, where the run-time is a time during which the node with no hard-coded conversation logic is being executed (Spec., 11:16-18; 18:4-20:2).

## **VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL**

- A. Claims 1, 11-14, 16-18, 27, and 31 were rejected under 35 U.S.C. § 102(a) as anticipated by Kuno (“Conversions + Interferences = Business Logic”) in view of Stewart (U.S. Patent No. 7,051,072).<sup>1</sup>**
- B. Claims 2, 3, 5-7, 28, and 30 were rejected under 35 U.S.C. § 103(a) as unpatentable over Kuno in view of Czerwinski (“An Architecture for a Secure Service Discovery Service”) and Stewart.<sup>2</sup>**

## **VII. ARGUMENT**

The claims do not stand or fall together. Instead, Appellant presents separate arguments for various independent and dependent claims. Each of these arguments is separately argued below and presented with separate headings and sub-headings as required by 37 C.F.R. § 41.37(c)(1)(vii).

- A. Claims 1, 11-14, 16-18, 27, and 31 were rejected under 35 U.S.C. § 102(a) as anticipated by Kuno (“Conversions + Interferences = Business Logic”) in view of Stewart (U.S. Patent No. 7,051,072).**

### **1. Claims 1, 27.**

Independent claim 1 recites a method for selecting a conversation logic at run-time for a work flow definition that includes at least one node with no hard-coded conversation logic, comprising:

- a) maintaining a conversation logic repository that includes plural conversation logic, wherein each of the plural conversation logic is external to the workflow definition, and wherein each of the plural conversation logic specifies a corresponding set of operations to be performed on a respective service;
- b) when executing the node with no hard-coded conversation logic, dynamically discovering, by a computer, a service associated with the node with no hard-coded conversation logic, wherein the discovered service is selected from among plural services;

---

<sup>1</sup> Page 3 of the 10/27/2010 Office Action incorrectly rejected cancelled claims 19 and 29 over Kuno and Stewart.

<sup>2</sup> Page 9 of the 10/27/2010 Office Action incorrectly rejected cancelled claim 9 over Kuno, Stewart, and Czerwinski.

- c) selecting one of the plural conversation logic in the conversation logic repository based on the discovered service; and
- d) dynamically plugging in the determined selected conversation logic into the node at run time in the computer, wherein the run time is a time during which the node with no hard-coded conversation logic is being executed.

It is respectfully submitted that the obviousness rejection of claim 1 over Kuno over Stewart is erroneous.

To make a determination under 35 U.S.C. § 103, several basic factual inquiries must be performed, including determining the scope and content of the prior art, and ascertaining the differences between the prior art and the claims at issue. *Graham v. John Deere Co.*, 383 U.S. 1, 17, 148 U.S.P.Q. 459 (1965). Moreover, as held by the U.S. Supreme Court, it is important to identify a reason that would have prompted a person of ordinary skill in the art to combine reference teachings in the manner that the claimed invention does. *KSR International Co. v. Teleflex, Inc.*, 127 S. Ct. 1727, 1741, 82 U.S.P.Q.2d 1385 (2007).

The Examiner conceded that Kuno fails to disclose clause d) of claim 1 set forth above. 10/27/2010 Office Action at 4. Instead, the Examiner cited Stewart as purportedly disclosing the claimed subject matter. *Id.* at 4-5. Specifically, the Examiner cited the following passages of Stewart: column 19, line 19 – column 20, line 54; column 10, lines 20-31. *Id.*

It is respectfully submitted that Stewart fails to disclose or hint at the claimed subject matter conceded to be missing from Kuno. The cited passage in columns 19 and 20 of Kuno has three sections: (1) a section describing an integration server, (2) a section describing business protocol and logic plug-ins, and (3) a section describing logic plug-ins. The section describing the integration server refers to a workflow server that is initialized at design time to have a workflow. Stewart, 19:21-23. Initialization of the workflow server to have a workflow (at design time) is done by creating a new workflow template and using the template to define an

XML document. *Id.*, 19:23-25. A set of business operations can be stored with the workflow templates. *Id.*, 19:26-28.

At runtime, workflow instances based on the workflow templates can be started. *Id.*, 19:29-31. The workflow server executes these instances to effect the workflow. *Id.*, 19:31-32.

It is noted that claim 1 recites dynamically plugging in the determined selected conversation logic into the node (included as part of a workflow definition) at runtime in a computer. However, it is clear that the section describing the integration server in column 19 of Stewart provides no teaching or hint of dynamically plugging in any conversation logic into a node of its workflow template. Even though the Examiner does not explicitly state, it appears that the Examiner is likely considering the workflow template discussed in this section of Stewart as corresponding to the “workflow definition” recited in claim 1. Thus, even if the workflow definition can be considered to have nodes, it is clear that the section describing the integration server of Stewart provides no teaching or hint of dynamically plugging in any conversation logic into any node of the workflow template.

The other sections in the passage in columns 19-20 cited by the Examiner refer to logic plug-ins. As explained by Stewart, logic plug-ins allow a c-space owner to add unique functionality to a c-space. Stewart, 19:61-63. As explained in Stewart, “c-space” refers to a collaboration space. *Id.*, 14:53. The c-space is an abstraction supporting a single business model, business message protocols, a secure message space, security policies, quality of service policies, and a registered set of business trading partners. *Id.*, 14:56-60. However, it is clear that the collaboration space, in which trading partners are able to collaborate, does not constitute a workflow definition as recited in claim 1. Note that, according to claim 1, each of the plural conversation logic is external to the workflow definition. Thus, claim 1 makes clear that the



workflow definition is distinct and separate from the plural conversation logic. Thus, to the extent that the collaboration space of Stewart is considered to implement conversations between trading partners, it is clear that the collaboration space of Stewart cannot be the workflow definition of claim 1. Thus, adding logic plug-ins to the collaboration space of Stewart does not provide any hint of dynamically plugging in a selected conversation logic into the node of a workflow definition at runtime.

The other passage of Stewart cited by the Examiner is column 10, lines 20-31. This passage of Stewart describes pluggable business logic and protocol support. The passage refers to a plug-in architecture for dynamic and intelligent routing of messages, to enable market makers to design and implement business rules that meet specific needs. However, there is no hint in the column 10 passage of Stewart regarding dynamically plugging in any conversation logic into a node of a workflow definition at runtime.

Since Stewart does not provide any teaching or hint of claimed subject matter conceded by the Examiner to be missing from Kuno, it is respectfully submitted that the hypothetical combination of Kuno and Stewart would not have led to the subject matter of claim 1.

Moreover, in view of the significant differences between the claimed subject matter and the teachings of Kuno and Stewart, no reason existed that would have prompted a person of ordinary skill in the art to combine the teachings of the references to achieve the subject matter of claim 1. In the Final Rejection, with respect to clause b) of claim 1, the Examiner cited § 4.1 of Kuno as purportedly disclosing “executing the node with no hard-coded conversation logic.” 10/27/2010 Office Action at 4. Specifically, the Examiner alleged that the e-service client in this passage of Kuno is not hard-coded with conversation logic. *Id.* Thus, the Examiner appears to have equated the “node” of claim 1 with the e-service client of Kuno. As explained by Kuno, an

e-service client is the client that interacts with an e-service. *See, e.g.,* Kuno, § 2, p. 3, ¶ 3. Section 4.1 of Kuno states that decoupling conversation logic from business logic on the client side increases the flexibility of the client by allowing the client to interact dynamically with services even if their conversation policies do not match exactly. *Id.*, § 4.1, p. 10, ¶ 1.

Note that the “node” recited in claim 1 is included in a workflow definition. Note also that clause d) of claim 1 recites dynamically plugging in a determined selected conversation logic into such node at runtime. An e-service client cannot constitute a node of a workflow definition, and thus equating the “node” of claim 1 with the e-service client of Kuno constitutes error. Moreover, it is clear that the teachings of Kuno would not have provided any hint of dynamically plugging in a selected conversation logic into an e-service client. Thus, even if Stewart were to disclose the dynamic plugging of a determined selected conversation logic into a node of a workflow definition at runtime (which Stewart does not as explained above), there would have been no reason to incorporate the teachings of Stewart into Kuno.

Therefore, it is clear that the obviousness rejection of claim 1 and its dependent claims over Kuno and Stewart is erroneous.

Reversal of the final rejection of the above claims is respectfully requested.

## **2. Claims 11-14, 16-18, 31.**

Independent claim 11 is also non-obvious over Kuno and Stewart for similar reasons as stated above with respect to claim 1. Specifically, with respect to claim 11, contrary to the assertion of the Examiner, Stewart fails to disclose the following element of claim 11 conceded by the Examiner to be missing from Kuno:

wherein the dynamic conversation logic selection mechanism is configured to further dynamically plug in the selected conversation logic into the node at the

run-time, where the run-time is a time during which the node with no hard-coded conversation logic is being executed.

Moreover, for reasons similar to those stated above with respect to claim 1, no reason existed that would have prompted a person of ordinary skill in the art to combine the teachings of Kuno and Stewart to achieve the subject matter of claim 11.

Therefore, the obviousness rejection of claim 11 and its dependent claims is erroneous.

**Reversal of the final rejection of the above claims is respectfully requested.**

**B. Claims 2, 3, 5-7, 28, and 30 were rejected under 35 U.S.C. § 103(a) as unpatentable over Kuno in view of Czerwinski (“An Architecture for a Secure Service Discovery Service”) and Stewart.**

**1. Claims 2, 30.**

In view of the allowability of base claims 1 and 11 over Kuno and Stewart, the obviousness rejection of dependent claims 2 and 30 over Kuno, Czerwinski, and Stewart has been overcome.

Reversal of the final rejection of the above claims is respectfully requested.

**2. Claims 3, 5-7, 28.**

Independent claim 3 recites a method for selecting a conversation logic at run-time comprising:

maintaining a conversation logic repository that includes plural conversation logic, wherein each of the plural conversation logic specifies a corresponding set of operations to be performed on a respective service;

at run-time, sending a service selection query to an electronic services platform or other service broker, wherein the service selection query is for selecting a service from among plural services, wherein the run-time is a time during which a node of a workflow definition is being executed, where the node is with no hard-code of conversation logic;

receiving, by a computer, a returned service identifier corresponding to the selected service;

selecting, by the computer, a conversation logic from among the plural conversation logic in the conversation logic repository based on the returned service identifier; and

dynamically plugging in the selected conversation logic into the node with no hard-coded conversation logic at the run-time.

In the rejection of claim 3, the Examiner conceded that Kuno and Czerwinski fail to disclose the following element of claim 3:

dynamically plugging in the selected conversation logic into the node with no hard-coded conversation logic at the run-time.

10/27/2010 Office Action at 10-11. Instead, the Examiner cited Stewart as purportedly disclosing this specific claim subject matter. As explained above in connection with claim 1, Stewart clearly provides no hint of this subject matter conceded to be missing from other references. Therefore, even if Stewart, Czerwinski, and Kuno could be hypothetically combined, the hypothetical combination of the references would not have led to the subject matter of claim 3.

Moreover, since no reason existed that would have prompted a person of ordinary skill in the art to combine the teachings of Kuno and Stewart, as discussed above in connection with claim 1, a person of ordinary skill in the art would also not have been prompted to combine the teachings of Kuno, Czerwinski, and Stewart to achieve the subject matter of claim 3.

Therefore, the obviousness rejection of claim 3 and its dependent claims is also clearly erroneous.

Reversal of the final rejection of the above claims is respectfully requested.

## CONCLUSION

In view of the foregoing, reversal of all final rejections and allowance of all pending claims is respectfully requested.

Respectfully submitted,

Date: March 24, 2011

/Dan C. Hu/

Dan C. Hu  
Registration No. 40,025  
TROP, PRUNER & HU, P.C.  
1616 South Voss Road, Suite 750  
Houston, TX 77057-2631  
Telephone: (713) 468-8880  
Facsimile: (713) 468-8883

## **VIII. APPENDIX OF APPEALED CLAIMS**

Claims 4, 8-10, 15, 19-26, and 29 have been cancelled.

The claims on appeal are:

- 1 1. A method for selecting a conversation logic at run-time for a workflow definition that  
2 includes at least one node with no hard-coded conversation logic, the method comprising the  
3 steps of:
  - 4 a) maintaining a conversation logic repository that includes plural conversation  
5 logic, wherein each of the plural conversation logic is external to the workflow definition, and  
6 wherein each of the plural conversation logic specifies a corresponding set of operations to be  
7 performed on a respective service;
  - 8 b) when executing the node with no hard-coded conversation logic, dynamically  
9 discovering, by a computer, a service associated with the node with no hard-coded conversation  
10 logic, wherein the discovered service is selected from among plural services;
  - 11 c) selecting one of the plural conversation logic in the conversation logic repository  
12 based on the discovered service; and
  - 13 d) dynamically plugging in the determined selected conversation logic into the node  
14 at run time in the computer, wherein the run time is a time during which the node with no hard-  
15 coded conversation logic is being executed.

1     2.     The method of claim 1  
2             wherein the step of when executing the node with no hard-coded conversation logic,  
3     dynamically discovering the service associated with the node with no hard-coded conversation  
4     logic includes the steps of:  
5             sending a service selection query;  
6             in response to the service selection query, selecting a service from among the plural  
7     services based on a service selection rule;  
8             receiving a service reference; and  
9             wherein the step of selecting the conversation logic in the conversation logic repository  
10    based on the discovered service further includes the step of  
11             using the service reference to select the one conversation logic from the plural  
12    conversation logic for the selected service.

1     3.     A method for selecting a conversation logic at run-time comprising the steps of:  
2             maintaining a conversation logic repository that includes plural conversation logic,  
3     wherein each of the plural conversation logic specifies a corresponding set of operations to be  
4     performed on a respective service;  
5             at run-time, sending a service selection query to an electronic services platform or other  
6     service broker, wherein the service selection query is for selecting a service from among plural  
7     services, wherein the run-time is a time during which a node of a workflow definition is being  
8     executed, where the node is with no hard-code of conversation logic;  
9             receiving, by a computer, a returned service identifier corresponding to the selected  
10    service;  
11             selecting, by the computer, a conversation logic from among the plural conversation logic  
12    in the conversation logic repository based on the returned service identifier; and  
13             dynamically plugging in the selected conversation logic into the node with no hard-coded  
14    conversation logic at the run-time.

1     5.     The method of claim 3 wherein a particular one of the plural conversation logic is for the  
2     exclusive use of a given one of the plural services.

1 6. The method of claim 5 wherein another of the plural conversation logic is shared by two  
2 or more of the plural services.

1 7. The method of claim 3 wherein the conversation logic is not defined in the workflow  
2 definition at process definition time, the workflow definition defining a procedure that executes  
3 services.

1 11. A system for dynamically selecting a conversation logic at run-time for a workflow  
2 definition that includes at least one node with no hard-coded conversation logic comprising:  
3 a workflow engine for processing a workflow definition;  
4 a conversation logic repository that includes plural conversation logic that are external to  
5 the workflow definition, wherein each of the plural conversation logic specifies a corresponding  
6 set of operations to be performed on a respective service;  
7 an engine configured to select one of plural services for execution of the node with no  
8 hard-coded conversation logic; and  
9 a dynamic conversation logic selection mechanism configured to receive a service  
10 identifier that is associated with the selected service at run-time, and based on the service  
11 identifier to select a conversation logic from the plural conversation logic for interacting with the  
12 selected service at the run-time, and  
13 wherein the dynamic conversation logic selection mechanism is configured to further  
14 dynamically plug in the selected conversation logic into the node at the run-time, where the run-  
15 time is a time during which the node with no hard-coded conversation logic is being executed.

1 12. The system of claim 11 further comprising:  
2 a source for the plural services, wherein the source is configured to discover the selected  
3 service based on a service selection rule;  
4 wherein the dynamic conversation logic selection mechanism is configured to select the  
5 conversation logic from the plural conversation logic based on the discovered service.

1 13. The system of claim 12 wherein the source is one of a service broker, a service  
2 marketplace, and an e-services platform.



- 1 14. The system of claim 11, wherein only services that have a conversation protocol  
2 compatible with one of the conversation logic available in the repository are considered for  
3 selection for execution of the node.
- 1 16. The system of claim 11 wherein a particular one of the plural conversation logic is for the  
2 exclusive use of a given one of the plural services.
- 1 17. The system of claim 16 wherein another of the plural conversation logic is shared by two  
2 or more of the plural services.
- 1 18. The system of claim 11 wherein the selected conversation logic is not defined in the  
2 workflow definition at process definition time.
- 1 27. The method of claim 1, wherein different ones of the plural conversation logic are  
2 compatible with different ones of the plural services, and wherein selecting one of the plural  
3 conversation logic comprises selecting a conversation logic that is compatible with the  
4 discovered service.
- 1 28. The method of claim 3, wherein different ones of the plural conversation logic are  
2 compatible with different ones of the plural services, and wherein selecting one of the plural  
3 conversation logic comprises selecting a conversation logic that is compatible with the selected  
4 service.

1 30. The system of claim 11, wherein the engine to select one of the plural services is  
2 configured to:  
3 submit a service selection query to an electronic services platform to perform selection of  
4 the selected service from the plural services.

1 31. The system of claim 11, wherein different ones of the plural conversation logic are  
2 compatible with different ones of the plural services, and wherein selecting one of the plural  
3 conversation logic comprises selecting a conversation logic that is compatible with the selected  
4 service.

## **IX. EVIDENCE APPENDIX**

None.

**X.     RELATED PROCEEDINGS APPENDIX**

None.